Linux for Oracle specialists

Frits Hoogland

DBA Symposium 18 November 2008
WHO AM I?

- Frits Hoogland
- Working with oracle software since 1996
- Blogging at http://frits.homelinux.com
- Email: frits@frits.homelinux.com
- Oracle DB, Application server(s), EBS
- Tomcat, Jboss, Websphere
- Postgresql, enterprise DB, MySQL
- Apache
- Technical (network) security
AGENDA

- Poll
- History, redhat and oracle
- Linux 32 and 64 bit
- Access to linux systems
- X
- IO
- Handy tools: logrotate
- Handy tools: sar
- Handy tools: sosreport
- Handy tools: mii-tool/ethtool
- Handy tools: iperf
POLL

- Anyone using linux for corporate systems with oracle?
- Coming from:
  - Unix
  - Mainframe
  - Windows
- Tidbits:
  - 32 bit
  - 64 bit
- Version:
  - RHEL3
  - RHEL4
  - RHEL5
What is Linux?

- A unix like kernel
- Created and conceived in 1991 by a Finnish student called Linus Torvalds
- A kernel is not an complete operating system
Why Linux?
- A project to study the intel 80386 processor

Close relationship with Minix (Tanenbaum et al)
- Restricted use of Minix source
- Linux was depended upon Minix userspace
HISTORY, REDHAT AND ORACLE

- GNU
  - Free software mass collaboration project
  - Richard Stallman
  - Januari 1984
  - “GNU’s Not Unix”
  - GNU General Public License “GPL”
HISTORY, REDHAT AND ORACLE

• GNU software includes:
  – bash
  – Autoconf
  – Binutils
  – Bison
  – Emacs
  – Gcc
  – Gnome
  – Libstdc++
HISTORY, REDHAT AND ORACLE

- Minix 16-bit design at that time
  - Linux specifically meant for 32 bit intel 80386
  - Debate/critisism about kernel design

- Linux could benefit from GNU userspace tools

- GNU could benefit from Linux kernel

- Linux released under GPL in 1992
  - version 0.12
  - With GNU userland tools
HISTORY, REDHAT AND ORACLE

- GNU General Public License; GNU GPL; GPL

  - Free software license
  - Copyleft license

  - Free for anyone to use
  - Free for anyone to modify
  - Modifications need to be made publically available
  - GPL’ed source can not be made close source
  - Source needs to be available

This is my interpretation, I am not a lawyer!
HISTORY, REDHAT AND ORACLE

• Distribution / “Distro”
  - Linux kernel
  - GNU tools
  - X window system
  - Distribution specific things
    • Startup scripts
    • Package manager
  - Much more...
Distributions

- Anyone is free to make a distribution...

- Commercial: RedHat, SUSE, Ubuntu, Mandriva
- Community: Debian, Gentoo
- Single person: Slackware

- Amongst the approx. ~ 345 distributions
Back to the Oracle connection...

- Supported Distro’s for Oracle products:
  - RedHat enterprise Linux (RHEL)
  - SuSE Linux Enterprise Server (SLES)
  - Asianux

- See metalink “certify” tab for specific details
The RHEL – OEL connection

- RHEL *needs* to make source available

- Source can be recompiled to make RHEL clone:
  - CentOS (most popular) – [http://www.centos.org](http://www.centos.org)
  - Whitebox enterprise Linux – [http://whiteboxlinux.org](http://whiteboxlinux.org)
  - StartCom Linux – [http://www.startcom.org](http://www.startcom.org)

- Oracle Enterprise Linux – [http://linux.oracle.com](http://linux.oracle.com)
HISTORY, REDHAT AND ORACLE

- The RHEL – OEL connection
  - OEL is created from the source of RHEL
    - Different artwork
    - Compatible at source and binary level
    - Different online repository (up2date)
    - Using oracle’s up2date repository with RHEL install!
    - Oracle patches
HISTORY, REDHAT AND ORACLE

• The RHEL – OEL connection

• Conclusion:
  – RHEL and OEL are made to be fully compatible
  – New versions are made from RHEL source
  – Oracle can patch something, and donate it to linux community (for version 5):
    • Kernel: patch for bug with bonding ethernet devices
    • Kernel-utils: Updated firmware from Intel
LINUX 32 AND 64 BIT

- Linux 32 bit versus Linux 64 bit
  - Processor address bus width
  - 32 bit Linux can be installed on 64 bit processor
  - 64 bit Linux can NOT be installed on 32 bit processor
- Why? Memory addressing!
  - $2^{32} = 4,294,967,296 = 4\text{ GB}$
  - $2^{64} = 18,446,744,073,709,551,616 = 16\text{ EB}$
LINUX 32 AND 64 BIT

• RHEL4 / 32 bit kernel types:
  – Standard/UP kernel: kernel-2.6.9-78.EL.i686.rpm
    • 3GB userspace, 1GB kernelspace
    • 4GB addressable
  – SMP kernel: kernel-smp-2.6.9-78.EL.i686.rpm
    • 3GB userspace, 1GB kernelspace
    • 16GB addressable
  – Hugemem kernel: kernel-hugemem-2.6.9-78.EL.i686.rpm
    • 4GB userspace, 4GB kernelspace *
    • 64GB addressable
LINUX 32 AND 64 BIT

- 16 GB ?!
- 64 GB ?!

- But that is *physically* impossible!
  (again: $2^{32} = 4'294'967'296 = 4$ GB)
LINUX 32 AND 64 BIT

• PAE
  – Physical Address Extension

• Processor option:

  [root@test ~]# cat /proc/cpuinfo
  ...
  flags           : fpu vme de pse tsc msr paes mce cx8 apic
  mtrr pge mca cmov pat pse36 clflush mmx fxsr sse
  sse2 constant_tsc monitor
LINUX 32 AND 64 BIT

- PAE increases the number of bits of the address bus to 36!
- $2^{36} = 68'719'476'736 = 64$ GB
- But it’s not that easy...
LINUX 32 AND 64 BIT

• Virtual addressing in userspace is not modified
  – 32-bits addresses
  – 32-bits -> 4GB maximum!

• So all limits still apply
  – Even with PAE

• Need special call to address memory using PAE
LINUX 32 AND 64 BIT

- PAE & Oracle database
  - “VLM” / “Very Large Memory”
  - USE INDIRECT DATA BUFFERS = true
    - Effectively places the buffercache in PAE memory
  - Usage of ramfs (or shmfs or tmpfs): /dev/shm
  - Following parameters are incompatible with VLM:
    - SGA_TARGET
    - DB CACHE SIZE (only DB BLOCK BUFFERS allowed)
    - DB_xK_CACHE_SIZE
    - MEMORY_TARGET

This is essentially a 32 bit issue, same with other 32-bit OS’es
LINUX 32 AND 64 BIT

- PAE & Oracle database
  - Only for putting buffercache beyond 4GB
  - All other structures remain in 32-bit addressable memory
    - Which means 4GB* limit still applies to those
LINUX 32 AND 64 BIT

- RHEL5 / 32 bit kernel types:
  - Standard kernel: kernel-2.6.18-92.el5.i686.rpm
    - 3GB userspace, 1GB kernelspace
    - 4GB addressable
  - PAE kernel: kernel-PAE-2.6.18-92.el5.i686.rpm
    - 3GB userspace, 1GB kernelspace
    - 64GB addressable (16GB supported?)
LINUX 32 AND 64 BIT

- **RHEL4 / 64 bits**
  - Kernel-2.6.9-78.EL.x86_64.rpm
    - Standard kernel for UP systems
  - Kernel-smp-2.6.9-78.EL.x86_64.rpm
    - Kernel for systems with up to 8 CPU’s
  - Kernel-largesmp-2.6.9-78.EL.x86_64.rpm
    - Kernel for systems with up to 512 CPU’s

- **RHEL5 / 64 bits**
  - Kernel-2.6.18-92.el5.x86_64.rpm

- **For version 4 and 5 kernel current limit is 256GB**
LINUX FOR ORACLE SPECIALISTS – FRITS HOOGLAND

LINUX 32 AND 64 BIT

- General recommendations:
  - 32 bit hardware is deprecated
  - 64 bit hardware is not excessively more expensive
  - Use 64 bit O/S whenever possible
  - Only use 32 bit O/S when no other options exist
    - It’s possible to use 32 bit software on 64 bit O/S!

- It *is* easy/likely to encounter memory scaling problems with 32-bit OS’es!!
ACCESS TO LINUX SYSTEMS

- Access & usage of linux(/unix) systems

- Common methods: *(poll: who uses)*
  - Telnet
  - Rsh/Rlogin/R*
  - FTP
  - XDMCP
  - VNC (?)
  - Ssh
  - Others?
ACCESS TO LINUX SYSTEMS

• Telnet
  – Does NOT encrypt any data, including password
  – No authentication to detect MITM attacks
  – Most telnet daemons have vulnerabilities

  – **STRONGLY** advised NOT to use it
  – I would call telnet usage blamable neglect

  – Even persistent telnet users (networking equip.) have ssh/encrypted access nowadays.
ACCESS TO LINUX SYSTEMS

- Rsh / rlogin
  - Does NOT encrypt any data, including password
  - No authentication to detect MITM attacks
  - Very weak authentication
  
  **STRONGLY** advised NOT to use it
  - I would call rsh/rlogin usage blamable neglect
ACCESS TO LINUX SYSTEMS

• FTP
  - Does NOT encrypt any data, including password
  - No authentication to detect MITM attacks
  - Still very popular and standard
  - Username/password easy sniff able
  - Usage defendable if files to be transferred are non-sensitive and non-valueable.
ACCESS TO LINUX SYSTEMS

- XDMCP
  - X Display Manager Control Protocol
  - The client sends a broadcast on UDP/177
  - Servers with XDMCP enabled send a ‘WILLING’ packet to sender
  - Not enabled in modern X implementations
    - Unencrypted
    - Not secure
    - DOS attacks using XDMCP possible
ACCESS TO LINUX SYSTEMS

- **VNC**
  - Virtual Network Computing
  - Developed by the Olivetti and Oracle research lab
    - 1999: acquired by AT&T, 2002: closed
  - The VNC protocol (RFB) is not a secure protocol
    - Passwords are not sent in plain text
    - Sensitive to brute-force attacks
    - Got known vulnerabilities
  - Continued by realvnc, ultravnc, tightvnc, etc.
The default VNC server uses its own username/password file.

- Upon login in VNC, an already started and authenticated session is continued

Known issues include:

- Hard to start vnc servers at startup
- Network sockets can get stuck during stop/start, resulting in non-usuable ports
- Single keystrokes resulting in repeating characters

Not suitable as primary access method “(?)”
- Advice: better (stable/secure/easy) solutions exist
ACCESS TO LINUX SYSTEMS

- SSH
  - Secure SHell
  - most common implementation: openssh
    - Available on all popular Linux and Unix distributions
    - Available on Windows with Cygwin
    - The most popular implementation since 2005
  - Has file transfer module (SFTP)
  - Can ‘tunnel’ X traffic through the session
ACCESS TO LINUX SYSTEMS

- SSH uses public-key cryptography to authenticate the remote computer and allow the remote computer to authenticate the user, if necessary.

- Other functionality include:
  - Tunneling and forwarding (back or forth) of TCP ports
  - Backup, mirror or copy files secure using rsync
ACCESS TO LINUX SYSTEMS

Conclusion

- Security/secure access is important today
- Security will be more important in near future
- SSH is the only sensible method for remote usage of linux/unix servers
- SSH provides both access and file transfer
ACCESS TO LINUX SYSTEMS

- Many SSH clients exist
  - My favourite SSH client for windows is putty
    - http://www.chiark.greenend.org.uk/~sgtatham/putty/
  - My favourite SFTP client for windows is winscp
    - http://winscp.net

- Both are free
ACCESS TO LINUX SYSTEMS

- Nomachine
  - Open source version exists (freenx) but found I’ve too many bugs
  - No active processes when not used
  - Leverages openssh and openssh’s security
  - Suspendable sessions like VNC
    - If network/VPN drops, or need to continue at another place
  - *Extremely* fast, even over low bandwidth lines
ACCESS TO LINUX SYSTEMS

- Only uses 1 port; 22/tcp
  - Yes, that’s sshd
  - Sshd activates the session
- Free commercial version limited to 2 users/sessions
ACCESS TO LINUX SYSTEMS

D E M O
ACCESS TO LINUX SYSTEMS

- How does NoMachine work?

1. Ssh session is initiated
2. Logon using ssh certificate based authentication with nx user
3. Nxserver authenticates username and password with PAM
4. After authentication, a session is setup for username
X

- Graphical environment for Linux/Unix
- Needed for normal installation of Oracle software

Client/Server

- Client and server are reversed by X
  - The client starts a X server
  - Connection is made with the remote machine
  - The remote machine (the server ;-) is the client
X

- (microsoft) windows has no X server
  - Traditionally, commercial/expensive X servers were used on windows:
    - Reflection X, Exceed amongst others
  - Later, cygwin was used for X.
    - This required a significant installation of cygwin packages, but was free.
  - Now, we have Xming
    - Free, lightweight X server for windows (6MB)
Use (xming) X from a windows system

1. The remote ssh daemon must allow X11Forwarding:
   
   /etc/ssh/sshd_config:
   X11Forwarding yes

2. Start Xming locally
   By default, Xming listens on localhost
3. Enable X11 forwarding:
4. And start your session:
That’s all!

- No need to hassle with DISPLAY (!!)
- X is able to follow the session
  - When another host is accessed using ssh
    - Use ‘ssh -X’
    - X11Forwarding
X

- X and su-ing to other users
- I remember having troubles with X and su
  - DISPLAY lost
  - X authentication/cookies problems
1. Login on a box with X forwarding using putty:

```
login as: oracle
oracle@centos4's password:
[oracle@centos4 ~]$ echo $DISPLAY
localhost:11.0
[oracle@centos4 ~]$ xclock ← X app works
[oracle@centos4 ~]$
```
2. Now su to another user and start a X app again:

[oracle@centos4 ~] $ su –
Password:
[root@centos4 ~] # xclock

…..what happens?
X

WTF?! The xcloc displays !

- **PAM**
  - Plugable Authentication Modules
  - `/etc/pam.d/su`:
    ```
    ...
    session optional /lib/security/$ISA/
    pam_xauth.so
    ```

- Works only for sessions with X tunneling using ssh daemon, not on console
- Tested with CentOS 4&5
IO

- IO
  - Input and Output to devices
  - IO referred commonly to IO of blockdevices

- Normal / synchronous IO
  - pread64() / pwrite64() calls on linux
  - System call lasts entire read or write
  - This is done on oracle linux by default
IO

- Asynchronous IO
  - `io_submit()` / `io_getevents()` calls on linux
  - IO is submitted, process can continue
  - `filesystemio_options ‘setall’ or ‘asynch’`
    - Default ‘none’
  - Not beneficial for reading
    - Not used for reading, at least on linux
  - Can be beneficial for writing
IO

Synchronous write

\[ \text{pwrite64()} \]

\[ \text{wait} \]

\[ \text{IO call} \]

Session continues

Asynchronous write

\[ \text{io_submit()} \]

\[ \text{io_getevents()} \]

\[ \text{IO call} \]

Session continues
Asynchronous write

IO call

io_submit()  
io_getevents()
IO

- Demo!
  - Database with filesystemio_options = none
  - Database with filesystemio_options = asynch
IO

- **AIO**
  - Two proc entries for rhel4 & 5
    - `/proc/sys/fs/aio-max-nr`
      - System wide maximum number of aio contexts
    - `/proc/sys/fs/aio-nr`
      - Current number of used aio contexts
  - `aio-nr = aio-max-nr`
IO

- **Direct IO**
  - Optional flag (O_DIRECT) for opening file or device
  - Causes blocks **NOT** to be buffered at O/S layer
    - DIO has to be enabled explicitly on all platforms
      - Except windows
  - The symptom when O/S is caching and the database is called ‘double buffering’

  *Double buffering not always a bad thing!*
IO

- DIO is enabled for oracle when
  - filesystemio_options = setall or direct
- Detectable when stracing the startup of the database
- DEMO
IO

- noatime
  - Mount option for most filesystems
  - Means an entire filesystem is affected
  - Noatime = Access Time in Inode is not updated
    - An inode keeps 3 times: ctime, mtime and atime
  - This means every time an IO is done to a file, an extra IO is done to update the atime in the inode
  - noatime prevents update of atime
IO

- Example entry:

```
[root@centos4 ~]# cat /etc/fstab
...
/dev/sdc1 /oracle ext3 noatime 0 0
...
```
HANDY TOOLS: LOGROTATE

- Logrotate
  - Fully automated (log) files cleaner
  - Simple way of cleaning, without scripts
  - Available for any linux/unix platform
HANDY TOOLS: LOGROTATE

- Layout of an logrotate entry in /etc/logrotate.d:

  [root@test ~]# cat /etc/logrotate.d/sample
  /path/to/logfile /path/to/logfile {
    options
  }

HANDY TOOLS: LOGROTATE

• Example of logrotation for default apache:

```bash
[root@test ~]# cat /etc/logrotate.d/httpd
/var/log/httpd/*log {
  missingok
  notifempty
  sharedscripts
  postrotate
    /sbin/service httpd reload > /dev/null 2> /dev/null || true
  endscript
}
```
HANDY TOOLS: LOGROTATE

- Example of logrotation for oracle database:

  [root@test ~]# cat /etc/logrotate.d/oracle_database
/oracle/db/*/adump/*aud /oracle/db/*/bdump/*trc /oracle/db/*/cdump/core_* /oracle/db/*/udump/*trc {
  daily
  compress # compress after a day
  rotate 7 # remove after week
  missingok
}

(continued next slide)
HANDY TOOLS: LOGROTATE

/oracle/db/*/bdump/alert*log {
    copytruncate
daily
compress # compress after a day
rotate 7    # remove after week
missingok
}

(alert log file requires special handling
.....continued next slide)
/oracle/db/*/network/log/*log {
copytruncate
daily
compress # compress after a day
rotate 7    # remove after week
missingok
}

(listener log file requires special handling too
.....continued next slide)
HANDY TOOLS: LOGROTATE

/oracle/db/*/network/trace/*trc {
    daily
    compress # compress after a day
    rotate 7        # remove after week
    missingok
}

HANDBY TOOLS: SAR

- Sar (System Activity Reporter)
  - Collect, report or save system activity information
  - IO, Paging, Device activity, CPU activity, Runqueue, Process list, memory and swap, kernel tables, context switches, swapping, etc.
  - Very handy information for troubleshooting
  - Especially historically!
HANDY TOOLS: SAR

- Sar is not installed by default
  - Sar is installed by the `sysstat` rpm
  - The sysstat rpm is an oracle dependency
  - So it should be installed on a linux server with oracle products
HANDY TOOLS: SAR

- Historical sar data:
  - Available in /var/log/sa
  - Number of days available depended on:

[root@test ~]# cat /etc/sysconfig/sysstat
# How log to keep log files (days), maximum is a month
HISTORY=7
HANDY TOOLS: SAR

- Getting historical sar data

```
[root@test ~]# cd /var/log/sa
[root@test ~]# ls
sa20 sa21 sa24 sa25 sa26 sar20 sar25 sar26
[root@test ~]# sar -q -f sa25 -s 10:00:00 -e 11:00:00
```
<table>
<thead>
<tr>
<th>Time</th>
<th>Runq-sz</th>
<th>Plist-sz</th>
<th>LDavg-1</th>
<th>LDavg-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:00:01 AM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:10:01 AM</td>
<td>1</td>
<td>121</td>
<td>0.31</td>
<td>0.08</td>
</tr>
<tr>
<td>10:20:01 AM</td>
<td>1</td>
<td>121</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>10:30:01 AM</td>
<td>1</td>
<td>121</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>10:40:01 AM</td>
<td>0</td>
<td>121</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Average</td>
<td>1</td>
<td>121</td>
<td>0.04</td>
<td>0.03</td>
</tr>
</tbody>
</table>

[root@test ~]#
HANDY TOOLS: SOSREPORT

- Sosreport
  - Gather tool for redhat linux system information:
    - Settings
    - Logs
    - Hardware info
    - Disk/partitioning/space info
    - /proc/
    - Etc... (much more!)
  - Very useful for remote linux troubleshooting
HANDY TOOLS: MII-TOOL/ETHTOOL

- Both tools to display physical network card settings

[root@test ~]# mii-tool eth0
eth0: negotiated 100baseTx-FD, link ok
HANDY TOOLS: MII-TOOL/ETHTOOL

[root@test ~]# ethtool eth0
Settings for eth0
  Supported ports: [ TP MII ]
  Supported link modes: 10baseT/Half 10baseT/Full
                     100baseT/Half 100baseT/Full
  Supports auto-negotiation: Yes
  Advertised link modes: 10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
  Advertised auto-negotiation: Yes

(continued next slide)
HANDY TOOLS: MII-TOOL/ETHTOOL

Speed: 100Mb/s
Duplex: Full
Port: MII
PHYAD: 0
Transceiver: internal
Auto-negotiation: on
Current message level: 0x00000007 (7)
Link detected: yes
HANDY TOOLS: IPERF

- Iperf
  - Available in EPEL repository
    - This add’s the EPEL repository
  - iperf executable both client and server
    - iperf -s
    - iperf -c <hostname>
HANDYTOOLS: IPERF

[root@test ~]\# iperf -s

-----------------------------------------------------
Server listening on TCP port 5001
TCP windows size: 85.3 Kbyte (default

-----------------------------------------------------
HANDY TOOLS: IPERF

[root@test ~]# iperf -m -c localhost

-----------------------------------------------------
Client connecting to localhost, TCP port 5001
TCP window size: 49.2 Kbyte (default)
-----------------------------------------------------

[  3] local 127.0.0.1 port 44316 connected with 127.0.0.1 port 5001
[  3] 0.0-10.0 sec 3.92 Gbytes 3.36 Gbits/sec
[  3] MSS size 16384 bytes (MTU 16424 bytes, unknown interface)
HANDY TOOLS: IPERF

- Very useful for:
  - Detecting *real* network bandwidth
  - Investigating real MTU size